

07-20-00

A

Please type a plus sign (+) inside this box → ☐PTO/SB/05 (12/97)
Approved for use through 09/30/00. OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.**UTILITY
PATENT APPLICATION
TRANSMITTAL**

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.

K35A0625

Total Pages

First Named Inventor or Application Identifier

SCOTT T. HUGHES

Express Mail Label No.

EI266880190US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO:

Assistant Commissioner for Patents
Box Patent Application
Washington, DC 20231

1. ☒ Fee Transmittal Form
(Submit an original, and a duplicate for fee processing)
2. ☒ Specification [Total Pages **21**]
(preferred arrangement set forth below)
- Descriptive title of the Invention
 - Cross References to Related Applications
 - Statement Regarding Fed sponsored R & D
 - Reference to Microfiche Appendix
 - Background of the Invention
 - Brief Summary of the Invention
 - Brief Description of the Drawings (if filed)
 - Detailed Description
 - Claim(s)
 - Abstract of the Disclosure
3. ☒ Drawing(s) (35 USC 113) [Total Sheets **9**]
___ Formal ___X_ Informal
4. Oath or Declaration [Total Pages **1**]
- a. ☐ Newly executed (original or copy)
 - b. ☐ Copy from a prior application (37 CFR 1.63(d))
(for continuation/divisional with Box 17 completed)
[Note Box 5 below]
 - i. ☐ DELETION OF INVENTOR(S)
Signed statement attached deleting
inventor(s) named in the prior application,
see 37 CFR 1.63(d)(2) and 1.33(b).
5. ☐ Incorporation By Reference (useable if Box 4b is checked)
The entire disclosure of the prior application, from which a
copy of the oath or declaration is supplied under Box 4b,
is considered as being part of the disclosure of the
accompanying application and is hereby incorporated by
reference therein.

6. ☐ Microfiche Computer Program (Appendix)
7. Nucleotide and/or Amino Acid Sequence Submission
(if applicable, all necessary)
- a. ☐ Computer Readable Copy
 - b. ☐ Paper Copy (identical to computer copy)
 - c. ☐ Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

8. ☐ Assignment Papers (cover sheet & document(s))
9. ☐ 37 CFR 3.73(b) Statement ☐ Power of Attorney
(when there is an assignee)
10. ☐ English Translation Document (if applicable)
11. ☐ Information Disclosure Statement (IDS)/PTO-1449 ☐ Copies of IDS Citations
12. ☐ Preliminary Amendment
13. ☒ Return Receipt Postcard (MPEP 503)
(Should be specifically itemized)
14. ☐ Small Entity ☐ Statement filed in prior application,
Statement(s) Status still proper and desired
15. ☐ Certified Copy of Priority Document(s)
(if foreign priority is claimed)
16. ☐ Other:

17. If a CONTINUING APPLICATION, check appropriate box and supply the requisite information:

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No: _____/_____**18. CORRESPONDENCE ADDRESS**☐ Customer Number or Bar Code Label

(Insert Customer No. or Attach bar code label here)

or ☒ Correspondence address below

NAME	WESTERN DIGITAL CORPORATION				
	Milad G. Shara, Esq. - Reg. 39,367 <i>July 18, 2000</i>				
ADDRESS	8105 IRVINE CENTER DRIVE				
	PLAZA 3				
CITY	IRVINE	STATE	CALIFORNIA	ZIP CODE	92618
COUNTRY	U.S.A.	TELEPHONE	(949) 932-5676	FAX	(949) 932-5633

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

**METHOD OF AND CONTENT DELIVERY SERVER FOR DELIVERING
CONTENT TO A PERSONAL COMPUTER HAVING A DISK DRIVE WHICH
INCLUDES A NETWORK ADDRESS FOR THE CONTENT DELIVERY
SERVER AND A SERVER-CONTACTING PROGRAM**

Related Applications

The present application is related to the following patent applications, which are each incorporated by reference in their entirety: U.S. Patent Application No. 09/585,129 titled "SYSTEM AND METHOD OF RECEIVING ADVERTISEMENT CONTENT FROM ADVERTISERS AND DISTRIBUTING THE ADVERTISING CONTENT TO A NETWORK" filed 5/31/2000; U.S. Patent Application No. _____, titled "DISK DRIVE AND METHOD OF MANUFACTURING SAME INCLUDING A NETWORK ADDRESS AND SERVER-CONTACTING PROGRAM", filed 7/18/2000; U.S. Patent Application No. _____, titled "COMPUTER NETWORK AND CONNECTION METHOD FOR CONNECTING A PERSONAL COMPUTER AND A CONTENT DELIVERY SYSTEM USING A DISK DRIVE WHICH INCLUDES A NETWORK ADDRESS AND SERVER-CONTACTING PROGRAM", filed 7/18/2000; and U.S. Patent Application No. _____, titled "METHOD OF AND PERSONAL COMPUTER FOR DISPLAYING CONTENT RECEIVED FROM A CONTENT DELIVERY SERVER USING A DISK DRIVE WHICH INCLUDES A NETWORK ADDRESS FOR THE CONTENT DELIVERY SERVER AND A SERVER-CONTACTING PROGRAM", filed 7/18/2000.

Background of the Invention

Field of the Invention

The field of the invention relates to content delivery systems. More particularly, the field of the invention relates to delivering content via a content delivery server.

Background Information

A content data provider is a company that prepares, or transmits digital content, e.g., music, games, software utilities, advertisements. To be a successful content data provider it is necessary to obtain customer awareness with respect to the company itself, as well as to the content data provider's products.

Historically, there have been a number of advertising and distribution channels to promote digital content. For example, maintaining a website, direct mailing, and renting billboards are each alternative forms of advertisement that may be relied up to further promote brand recognition and other promote other advertising. However, as is

discussed below, each of these forms of advertisements has associated limitations and drawbacks.

Using the Internet, a content data provider can maintain a website whereby users can directly download and view advertising materials on their personal computer. However, one disadvantage of advertising on the Internet is that consumers need to know about the existence of the content data provider's website. In this regard, the content data provider needs to advertise its website or rely on search engines to provide the user with the Internet address of the user's company. Furthermore, to access the website the consumer needs to take an affirmative step and "navigate" to the web site. The term navigate is often referred to as the process of accessing a selected web page on the Internet. Disadvantageously, if the user never navigates to the content data provider's website, the content data provider is never given the opportunity to sell its products.

With respect to direct mail campaigns, many users never open, use, or install promotional CD packages that are sent. Another problem with direct mail campaigns is that due to the lack of ability to obtain feedback, it is difficult to gauge the successfulness of such campaigns.

Problems with billboards include: (i) they can be costly; (ii) there is no guarantee that individuals will see the billboard when driving by; (iii) it is difficult to determine the number of individuals that viewed the advertisement; and (iv) self evidently, there is not an "interactive" experience between the user and the billboard.

Thus, content data providers are in need of an advertisement and distribution channel that guarantees that personal computer users can receive their *content data*.

Summary of the Invention

One embodiment of the invention comprises a method of delivering content from a content delivery server to a personal computer system. The personal computer system includes a computing subsystem and a disk drive. The method comprises selecting a network address for the content delivery server, selecting a server-contacting program, storing the network address and the server-contacting program in the disk drive, and installing firmware in the disk drive to initiate execution of the server-

contacting program after the disk drive is connected to the computing subsystem in the personal computer system. Upon execution, the server-contacting program uses the network address to connect the personal computer system with the content delivery server. The content delivery server is configured for (i) receiving user information from the personal computer system while the personal computer system is connected to the content delivery server and (ii) delivering content to the personal computer system in response to the user information.

Another embodiment of the invention comprises a content delivery server for delivering content to a personal computer system. The personal computer system includes a computing subsystem and a disk drive. The content delivery server comprises means for communicating with a personal computer system having a disk drive. The disk drive comprises: a network address for identifying the content delivery server, a server contacting program, and a disk controller circuit. The disk controller circuit determines, upon the occurrence of a selected condition and after the disk drive is connected to the computing system, to initiate execution of the server-contacting program. Upon execution, the server-contacting program uses the network address to connect the personal computer system with the content delivery server. The content delivery server also comprises: (i) means for receiving user information from the personal computer system while the personal computer system is connected to the content delivery server; (ii) means for maintaining a database including the user information associated with the disk drive in the personal computer system; and (iii) means for delivering content to the personal computer system in response to the user information.

Brief Description of the Drawings

Figure 1 is a flowchart illustrating one embodiment of a content delivery process according to the present invention.

Figure 2 is a flowchart illustrating in further detail one embodiment of a manufacturing process that is shown in Figure 1, wherein the manufacturing process manufactures a disk drive.

Figure 3 is a flowchart illustrating in further detail an one embodiment of an installation process that is shown in Figure 1.

Figure 4 is a block diagram illustrating certain distribution channels for the disk drive that is manufactured by the process of Figure 2.

5 Figure 5 is a block diagram illustrating a network for distributing content data using the disk drive that is manufactured by the process of Figure 2.

Figure 6 is a block diagram illustrating certain elements of the disk drive that is manufactured by the process of Figure 2.

10 Figure 7 is a flowchart illustrating in further detail the installation process of Figure 3.

Figure 8 is a block diagram illustrating certain components of a content preparer.

Figure 9 is a flow chart of a process for delaying initiation of executing a server-contacting program until a selected condition is satisfied.

15 Detailed Description of the Embodiments of the Invention

Figure 1 is a flowchart illustrating one embodiment of a content delivery process. Alternatively, selected steps of the flowchart may be omitted and others may be added. Furthermore, depending on the embodiment, the ordering of the steps may be varied.

20 Starting at a step 100, a manufacturer manufactures a number of disk drives 516 (Figure 5). The process of manufacturing the disk drives 516 is described in further detail below with respect to Figure 2. However, in summary with respect to one embodiment of the manufacturing process, a first operating system 604, a server contacting program 608 and a network address 612 are stored in a protected area 628 in
25 the disk drives 516 (all shown in Figure 6). As is discussed further below, the network address 612 is a link or pointer to a server in the network. In one embodiment of the invention, the network address 612 is either a domain name, an URL, or an IP address. Furthermore, as part of the manufacturing process, firmware 616 on the disk drives 516 is adapted to detect when the disk drives 516 are connected in a personal computer
30 system and to initiate the loading of the first operating system 604. Upon being loaded, the first operating system 604, or an installation program executing under the first

operating system, installs the server-contacting program 608 on the personal computer system for use with a second operating system 620.

For convenience of description, the following text will refer to a single disk drive 516. However, it is to be appreciated that the same process may be performed with respect to manufacturing and using any number of disk drives. Furthermore, the disk drive 516 may be integrated with one of several personal computers 502A-502M (Figure 5). A personal computer may take the form of a laptop, desktop, palmtop, or other configuration suitable for a single user. Although the disk drive 516 may be integrated with any of the personal computers 502A-502M, for convenience of description, the following text assumes that the disk drive 516 will eventually be integrated with the personal computer 502M.

Referring again to Figure 1, at a step 104, the disk drive 516 is shipped to the consumer via one or more distribution channels. The disk drive 516 may be sold to an original equipment manufacturer (OEM) integrator, a distributor, a retailer, or directly to a consumer. Before sale to the consumer, the OEM integrator, the distributor and the retailer, may load a second operating system 620 and one or more application programs 624 on the disk drive 516. The second operating system 620 can include, for example: UNIX, LINUX, OS/2, PalmOS, Windows 98, Windows NT, and Windows CE. Since the first operating system 604 provides limited functionality, the second operating system 620 may be needed by the consumer to run the consumer's application programs.

Continuing to a step 108, the disk drive 516 is installed in the personal computer 502M. At the step 108, the disk drive 516 is physically connected to an adapter connector that is associated with a computing subsystem 518 (Figure 5) of the personal computer 502M. The second operating system 620 is then configured for operation with the disk drive 512. Furthermore, as will be described in further detail below with respect to Figure 3, upon the occurrence of one or more selected conditions, the disk drive 516 initiates the execution of the first operating system 604 which then installs the server contacting program 608 for operation with the second operating system 620. During the installation of the server-contacting program 608, the user may be requested

to provide user information, such as a name, an address, demographic information, and billing information.

Proceeding to a step 112, a content delivery server 512 is configured to receive user information, if the user agrees, from the personal computer 502M and to transmit content data to the personal computer 502M once the disk drive 516 is installed (described above with respect to step 108). At this step, the content delivery server 512 is connected to a network 520. The network address 612 in the disk drive 516 identifies the content delivery server.

Next, at step 116, the server contacting program 608 contacts the content delivery server 512. Furthermore, at the step 116, the content delivery server 512 stores the provided user information.

Moving to a step 120, the content delivery server 512 delivers content data to the personal computer 502M. The type of content data can vary greatly depending on the embodiment. Content data may include any type of digital file. For example, the content data can include: a content display program, a game, an entertainment program, a utility program, a product support program, entertainment data, advertisement data, music data, pictures or a movie. Furthermore, as non-limiting examples, the content data can be in any one of the following data formats: DHTML, .AVI, MPEG, QuickTime, a template, a static file, a Joint Photographic Experts Group (JPEG) file, an animated GIF, a Hypertext Markup Layout page, a .wav file, or a Macromedia flash file.

In one embodiment of the invention, the content data is displayed to the user during the boot process of the personal computer 502M. As defined herein, a boot process includes executing one or more programs to configure a personal computer to a state wherein the personal computer can execute an application program specified by the user after booting. In another embodiment of the invention, the content data is displayed to the user during a shutdown phase of the personal computer 502M. In yet another embodiment of the invention, the content data is displayed to the user during periods of low activity on the personal computer 502M.

Preferably, the personal computer 502M monitors the number of times that ads are displayed during boot and reports this number to a reporting server, e.g., the content

delivery server. Advantageously, content data providers can accurately determine the number of times that the advertisements are viewed by the consumer.

Figure 2 illustrates in further detail certain steps that occur in step 100 of Figure 1. It is to be appreciated that depending on the embodiment, selected steps of the flowchart may be omitted and that others may be added. Furthermore, depending on the embodiment, the ordering of the steps may be varied.

Starting at a step 200, a protected area 628 (Figure 6) is created on the disk drive 516. In one embodiment, the protected area 628 is written to and read from using proprietary commands that are provided by the manufacturer of the disk drive 516.

Next, at a step 204, the network address 612 for the content delivery server 512 is stored in the protected area 628. The network address 612 can include any type of identifier for an accessible device or content data in a network. Depending on the embodiment, more than one network address may be stored in the protected area 628.

Continuing to a step 208, the first operating system 604 and the server-contacting program 608 are stored in the protected area 628. Advantageously, since the first operating system 604, the server contacting program 608, and the network address 612 are stored in the protected area 628 in the disk drive 516, they cannot be easily deleted from the disk drive 516 by subsequent integration in the personal computer, if at all. The first operating system 604 can be any off-the-shelf or proprietary operating system. As non-limiting examples, the first operating system 604 can be any one of the following operating systems, for example: UNIX, LINUX, and Disk Operating System (DOS).

Moving to a step 212, firmware 616 is installed in the disk drive 516. Firmware 616 is generally microprocessor executable code which is stored in the disk drive in non-volatile memory and executed by a disk drive microprocessor (not shown). In one embodiment of the invention, the firmware 616 resides in a disk controller circuit 632. Disk controller circuit 632 comprises a microprocessor (not shown) suitable for executing firmware code 616. Under certain conditions, the firmware 616 is adapted to initiate execution of the first operating system 604 instead of the second operating system 620. In this regard, in one embodiment, of the invention, the firmware 616 is adapted to recognize when the computing subsystem 518 has requested a boot sector

address stored on the disk drive 516. The boot sector governs the behavior of the overall system during boot when control passes from POST Basic Input Output Services (BIOS) code. If the selected conditions, described below, are met and if the computing subsystem 518 has requested the boot sector address that is associated with the second
5 operating system 620, the firmware 616 substitutes a boot sector associated with the first operating system 604 for the requested sector. An exemplary is more fully described in U.S. Application No. 09/507,121 titled "Disk Drive for Selectively Satisfying a Read Request From a Host Computer For A First Valid Data Block With A Second Valid Data Block", filed on 2/17/2000 and which is incorporated by reference
10 herein in its entirety. In another embodiment, the firmware 616 may read the boot sector requested by the BIOS from disk and modify the code prior to transmitting the sector to the host computer system. The code modification may be accomplished by modifying data read from the sector using an algorithm or by "patching" the boot sector data with data stored elsewhere on the disk or embedded in the firmware code. In another
15 embodiment, the firmware 616 may bypass reading the boot sector from the disk and substitute an entire sector stored in non-volatile memory or otherwise embedded in the firmware code. The process flow then returns to step 104 of Figure 1 (discussed above).

Figure 3 is a flowchart illustrating one embodiment of a process for installing the server-contacting program 608 for operation with the second operating system 620.
20 Figure 3 illustrates in further detail certain steps that occur in steps 108 and 116 of Figure 1. It is to be appreciated that depending on the embodiment, selected steps of the flowchart may be omitted and that others may be added. Furthermore, depending on the embodiment, the ordering of the steps may be varied.

Starting at a step 300, the user boots the personal computer 502M. When the
25 system is powered on, a BIOS program is read from a read only memory in the computing subsystem 518. During its execution, the BIOS issues a plurality of read requests to the disk drive 516. As part of the boot process, the BIOS requests the disk drive 516 to provide a predetermined sector, i.e., the boot sector, to begin the process for loading an operating system.

30 Next, at a step 304, in response to each read request, the firmware 616 determines whether the read request is for the boot sector. During the step 304, the

firmware 616 attempts to activate the first operating system 604 at a point in time after the disk drive 516 is installed at the consumer's personal computer 502M. If the process of installing the server-contacting program 608 is initiated prematurely, the first operating system 604 and/or server-contacting program 608 may be overwritten, or alternatively, notice of the presence of the server-contacting program 608 may be disadvantageously disclosed. For example, often after manufacture, OEM integrator install and test application programs on the disk drive 516. If the process of installing the server contacting program 608 is initiated when the disk drive is in the possession of the OEM integrator, the OEM integrator may disadvantageously uninstall the server contacting program 608.

If the read request is for the boot sector, the firmware 616 determines whether a selected condition has been met and delays initiating execution of the server-contacting program unless the selected conditions are met. As shown in Fig. 9, the process 900 of determining when the selected condition is satisfied can be any one or more of a number of alternatives. The selected conditions can include, among other things, counting 902 a number of bootloads of the computing subsystem 518 or counting 906 until the number of power cycles performed by the disk drive 516 has exceeded a predetermined threshold. If either number exceeds the threshold, it can be assumed by the firmware 616 that the disk drive 516 is in the consumer's possession and is not in the hands of the OEM integrator or other individual that is in the shipping channel of the disk drive 516. This process is further described below with respect to Figure 7.

Another way of determining when the disk drive 516 has been installed with the personal computer 502M is to adapt the firmware 616 to always initiate execution of the first operating system 604. In one embodiment, a program is loaded by the first operating system 604 which reads (step 910) a system clock from the personal computer 502M. If a sufficient period of time has elapsed, it may be assumed that the disk drive 516 is in use by a consumer.

In another embodiment, firmware 616 may "scan" (step 912) data sectors on the disk to determine whether a bootable system image or operating system has been installed on the disk to determine if the drive has progressed through the channel to a point where the server-contacting program may be initiated. By employing knowledge

of conventional assignment of logical block addresses and a comparison of expected to recorded data, the firmware 616 may determine that a suitable operating environment exists for initiating execution of the server-contacting program. In one embodiment, the firmware 616 may comprise an embedded operating system such as Linux. When the selected condition is satisfied, the firmware 616 initiates (step 914) execution of the server-contacting program.

Moving to a step 308, the first operating system 604, or alternatively, an installation program running under the first operating system 604, installs the server-contacting program 608 for use with the user's personal computer 502M. As part of the installation process, the user may, depending on the embodiment, be requested to register to use the server-contacting program 608. In one embodiment of the invention, after being installed, the server contacting program 608 and the network address 612 each reside in the non-protected area of the disk drive 632. After being installed for use with the second operating system 620, the second operating system 620 automatically executes the server-contacting program 608 each time the second operating system 620 is executed.

Next, at a step 312, the second operating system 620 is executed. In one embodiment, to accomplish this step, a soft reboot is issued and during the reboot the firmware 616 initiates the loading of the second operating system 620. The second operating system 620 then executes the server-contacting program 608. In one embodiment of the invention, an exemplary server contacting program 608 is described in U.S. Application No. 5,913,040 to Rakavy, et al.

Continuing to a step 316, the server contacting program 608 reads the network address 612 that is stored in the disk drive 516. Moving to a step 320, the server-contacting program 608 uses the network address to contact the content delivery server 512 (the device identified by the network address 612). The process flow then returns step 120 of Figure 1 (discussed above).

Figure 4 is a block diagram illustrating certain distribution channels for the disk drive 516. After being manufactured, the disk drive 516 comprises the first operating system 604, the server contacting program 608, and the network address 612. Preferably, a display content engine 610 may be included. Display content engine 610

provides executable code for displaying a variety of digital content such as AVI, JPEG, MPEG, MP3 and other media standards. After manufacture, the disk drive 516 may then be shipped to a PC OEM integrator (block 404), to a distributor (block 408), or to a retailer (block 412). The distributor may further distribute the disk drive 516 to a VAR integrator or a second tier OEM (block 416). The retailer may sell the disk drive 516 to a consumer who installs the disk drive 516 for use with the consumer's personal computer. A process of installing the disk drive 516 is described above with respect to Figure 3. After installation, the consumer is then able to use the installed disk drive 516.

It is noted that the PC OEM integrator, the distributor, the retailer, the VAR integrator or second tier OEM, or the consumer may install the second operating system 620 and other application programs 624 on the disk drive 516.

Figure 5 is a block diagram illustrating an exemplary content delivery system 500. The content delivery system comprises a content delivery server 512 that is operably connected to personal computers 502A-502M via a network 520. In one embodiment of the invention, the content delivery server 512 comprises a plurality of geographic servers each having a unique IP address and having a common domain name. The content delivery server 512 is operably connected to a content preparer 514 which formats content data from the content provider 800 for use by the content delivery server 512. The operation of the content preparer 514 is described below in further detail with respect to Figure 8.

The personal computers 502A-502M each have a system unit 506, a display device 508, and an input device 510. The system unit 506 comprises a disk drive 516, a computing subsystem 518, and a display generator 522. In one embodiment, the computing subsystem 518 communicates with the disk drive 516 via a disk drive interface. The disk drive interface can be any standard or propriety storage interface. For example, the disk drive interface can be an Intelligent Drive Electronics (IDE) or AT Attachment (ATA) interface.

The computing subsystem 518 may include, among other elements, a processor, random access memory (RAM), read only memory (ROM), a BIOS stored in the ROM, and a system clock. The processor, may be one or more conventional general purpose

microprocessors such as a Pentium® processor, a MIPS® processor, a Power PC® processor, or an ALPHA® processor. The network 520 may include any type of electronically connected group of computers including, for instance, the following networks: Internet, Intranet, Local Area Networks (LAN) or Wide Area Networks (WAN). In addition, the connectivity to the network may be, for example, remote modem, Ethernet (IEEE 802.3), Token Ring (IEEE 802.5), Fiber Distributed Datalink Interface (FDDI) or Asynchronous Transfer Mode (ATM). As used herein, an Internet includes network variations such as public internet, a private internet, a secure internet, a private network, a public network, a value-added network, an intranet, and the like. The network 520 may include multiple network technologies including both high and low bandwidth connections. The network 520 may include non-persistent connections that are only available at intervals, *e.g.*, dial up telephone lines, and may also include persistent connections.

The input device 510 may be a keyboard, rollerball, pen and stylus, mouse, or voice recognition system. The input device 510 may also be a touch screen associated with the display device 508. The user may respond to prompts on the display by touching the screen. Textual or graphic information may be entered by the user through the input device 510.

Figure 6 is a representational block diagram illustrating certain elements of the disk drive 516. The disk drive 516 comprises a disk controller circuit 632 and a storage area 636. The disk controller circuit 632 comprises conventional disk controller components (not shown) such as a read channel device, formatter/controller, host interface logic and buffer controller as well as a microprocessor suitable for executing firmware 616 that is used to initiate the installation of the server contacting program 608 for use with the second operating system 620. Suitably, an ST10 microprocessor from ST microelectronics or an Intel x86 family microprocessor can be used although other equivalent devices or “cores” are available for integration in the disk controller circuitry. It is noted that the process of installing the server-contacting program 608 is described above with respect to Figure 3.

The storage area 636 comprises a protected area 628 and a non-protected area 632. During manufacture, the first operating system 604, the server-contacting program

608, and the network address 612 are stored in the protected area 628. After manufacture, the second operating system 620 and the application program 624 may be stored by an OEM or the consumer in the non-protected area 632. In one embodiment, the protected area may be implemented by partitioning the disk into two sets of addressable locations – a “protected set” and a “user accessible” set. A “protected” set of addressable locations is addressable only by sending commands or command protocols that are proprietary to the disk drive manufacturer. Data in the protected area may then only be accessed by the firmware or by a host executable program that is entrusted with knowledge of the proprietary commands. The user accessible area is available using standard access methods recognized by the disk drive such as ATA or SCSI standards. In another embodiment, the protected area may be created using standard ATA commands such as SET MAX and READ MAX which allow an area of the drive to be hidden from the host operating system. Other suitable protection may be employed to ensure that the programs in the protected area survive post manufacturing distribution channels.

Figure 7 is a flowchart illustrating in further detail certain steps of the installation process of Figure 3. It is to be appreciated that depending on the embodiment, selected steps of the flowchart may be omitted and that others may be added. Furthermore, depending on the embodiment, the ordering of the steps may be varied.

Starting at step 708, upon being accessed, the firmware 616 determines whether operating system substitution is disabled. The process of determining whether the operating system substitution is disabled is described above in further detail with respect to the description of step 304 of Figure 3, i.e., determining whether the selected conditions have been met. If substitution is disabled, the process flow proceeds to a step 728, wherein the firmware 616 performs the normal boot sequence and initiates the loading of the second operating system 620.

Referring again to step 708, if substitution is not disabled, the process flow proceeds to a step 712, wherein the firmware 606 reads the master boot record (conventionally the first logical block address) that is stored in the disk drive 516.

Next at a step 716, the firmware 616 determines whether the disk drive 516 contains a second operating system 620 that is compatible for use with the server contacting program 608. In one embodiment of the invention, to make this determination, the firmware 616 performs pattern matching of expected sequences in the master boot record. Depending on the embodiment, the boot record may contain: jump instructions to other routines, a version number of the operating system, a boot signature, or a magic number. By matching the values of these fields to expected values that are stored either in the firmware 616 or in the protected area 628, the firmware 616 determines whether the second operating 620 that was installed by the user or other party is supported for use with the server contacting program 608. In such event that the second operating system 620 is unknown and is not compatible for use with the server-contacting program 608, the process proceeds to the step 728.

Referring again to the step 716, if it is determined that the disk drive 516 contains a second operating system 620 that is compatible for use with server contacting program 608, the process proceeds to a step 720, wherein the firmware 616 determines whether the disk drive 516 is in a supported embodiment of system unit 506. At this step, the firmware 616 determines whether the system unit 506 is compatible for use with the first operating system 604 and the server-contacting program 608. In one embodiment of the invention, at this step, the firmware 616 analyzes the BIOS calls to the disk drive 516 to see if sequence corresponds to an expected pattern. If the BIOS calls of the computing system 518 conform to the expected pattern, the firmware 616 assumes that the system unit 506 is compatible for use with the first operating system 604 and the server contacting program 616.

If the disk drive 516 is in a supported embodiment of system unit 506, the process flow proceeds to a step 732 wherein the firmware 616 substitutes the boot record for the first operating system 604 with the boot record of the second operating system 620. This substitution initiates the loading of the first operating system 604.

Next, at a step 736, the first operating system 604, or an installation program executing under the first operating system 604, copies the server contacting program 608 and the network address 612 from the protected area 628 into the non-protected area 632.

Continuing to a step 740, the first operating system 604, or an installation program executing under the first operating system 604, modifies a system registry of the second operating system 620 to include a reference to the server contacting program. The system registry identifies application programs that should be automatically executed by the second operating system 620 after the second operating system 620 is executed.

Proceeding to a step 744, the firmware 616 disables substitution for subsequent booting of the second operating system 620. Next, at a step 748, the first operating system 604, or alternatively, an installation program executing under the first operating system 604, issues a soft boot, thereby initiating the execution of the second operating system 120 and, subsequently, the server contacting program 608.

Figure 8 is a block diagram illustrating in further detail certain components of one embodiment of the content preparer 514 of Figure 5. As is shown in Figure 8, the content preparer 5142 comprises a content repository 804, a content database 808, a publishing module 812, a publishing database 816, and a packaging module 820.

A content provider 800 can submit content data to the content preparer 514 via an on-line submission process that uses a web interface. As part of the submission process, a content provider 800 submits content data and meta-data characterizing the content data to the content preparer 514. The content data is stored in the content repository 804. The meta-data is stored in the content database 808. Periodically, a publishing module 812 extracts the meta-data from the content database 804 and the content from the content repository 804 and formats the material for distribution to the personal computer 502M. The reformatted data content is stored in a publishing database 816. Upon the occurrence of selected conditions, the packaging module 820 accesses the publishing database to determine which materials should be published and delivers the reformatted data content to the personal computer 502M.

Advantageously, the present invention allows disk drive manufacturers to provide content delivery software to users. Since the first operating system 604, the server contacting program 608, and the network address 612 are stored in a protected area 628, these files cannot be easily deleted by other entities in the distribution channels. Furthermore, by including the server-contacting program 608 with each

manufactured disk drive 516, the server-contacting program 608 is received by large numbers of personal computer users.

5

Table 1. Demographic characteristics of the study population	
Age (years)	65.5 ± 10.5
Gender	
Male	50 (50%)
Female	50 (50%)
Education (years)	12.5 ± 2.5
Marital status	
Married	40 (40%)
Single	10 (10%)
Widowed	40 (40%)
Divorced	10 (10%)
Occupation	
Retired	40 (40%)
Unemployed	10 (10%)
Employed	40 (40%)
Income (USD/month)	1,200 ± 300
Health status	
Good	40 (40%)
Fair	10 (10%)
Poor	40 (40%)
Chronic diseases	
Hypertension	30 (30%)
Diabetes	20 (20%)
Heart disease	10 (10%)
Stroke	5 (5%)
Other	15 (15%)

WHAT IS CLAIMED IS:

1 1. A method of delivering content from a content delivery server to a
2 personal computer system that includes a computing subsystem and a disk drive, the
3 method comprising:

4 selecting a network address for the content delivery server;

5 selecting a server-contacting program;

6 storing the network address and the server-contacting program in the disk
7 drive;

8 installing firmware in the disk drive to initiate execution of the server-
9 contacting program after the disk drive is connected to the computing subsystem
10 in the personal computer system, wherein execution of the server-contacting
11 program includes using the network address for connecting the personal
12 computer system to the content delivery server; and

13 configuring the content delivery server, the configuring comprising:

14 receiving user information from the personal computer system
15 while the personal computer system is connected to the content delivery
16 server; and

17 delivering content to the personal computer system in response to
18 the user information.

1 2. The method of Claim 1 wherein:

2 the disk drive includes a protected area; and

3 the network address and the server-contacting program are stored in the
4 protected area.

1 3. The method of Claim 1, wherein the firmware resides in a disk controller
2 circuit.

1 4. The method of Claim 1, wherein the firmware delays initiating execution
2 of the server-contacting program until a predetermined period has lapsed.

1 5. The method of Claim 1, wherein the personal computer system displays
2 the content during a boot sequence.

009648766-071800

1 6. The method of Claim 5, wherein the content is periodically changed
2 according to a presentation schedule.

1 7. The method of Claim 1, wherein the firmware delays initiating execution
2 of the server-contacting program until the firmware determines that a selected number
3 of monitored events exceeds a threshold.

1 8. The method of Claim 7, wherein the monitored events includes the
2 number of boot-ups in the computing subsystem.

1 9. The method of Claim 1, wherein the content is selected from the group
2 comprising: a content display program, a game, an entertainment program, a utility
3 program, entertainment data, advertisement data, and music data.

0961876-074800

1 10. A content delivery server for delivering content to a personal computer
2 system that includes a computing subsystem and a disk drive, the content delivery
3 server comprising:

4 means for communicating with a personal computer system, wherein the
5 personal computer system comprises:

6 a disk drive comprising:

7 a network address for identifying the content delivery
8 server; and

9 a server contacting program;

10 a disk controller circuit for, upon the occurrence of a
11 selected condition that is determined by the disk controller
12 circuit, initiating execution of the server-contacting program after
13 the disk drive is connected to the computing subsystem in the
14 personal computer system, wherein execution of the server-
15 contacting program includes using the network address for
16 connecting the personal computer system to the content delivery
17 server;

18 means for receiving user information from the personal computer system
19 while the personal computer system is connected to the content delivery server;

20 means for maintaining a database including the user information
21 associated with the disk drive in the personal computer system; and

22 means for delivering content to the personal computer system in response
23 to the user information.

1 11. The content delivery server of Claim 10, wherein the disk controller
2 circuit automatically initiates execution of the server-contacting program when the disk
3 drive is connected to the computing subsystem in the personal computer system.

1 12. The content delivery server of Claim 10, wherein the disk controller
2 circuit delays initiating execution of the server-contacting program until a
3 predetermined period has lapsed.

1 13. The content delivery server of Claim 10, wherein the firmware delays
2 initiating execution of the server-contacting program until the firmware determines that
3 a selected number of monitored events exceeds a threshold.

1 14. The content delivery server of Claim 13, wherein the monitored events
2 includes the number of boot-ups in the computing subsystem.

1 15. The content delivery server of Claim 10, wherein the personal computer
2 system displays the content during a boot sequence.

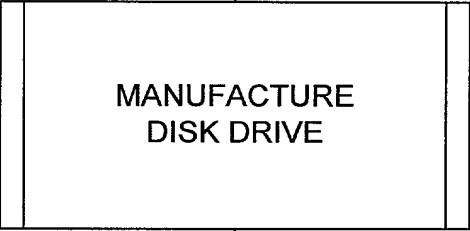
1 16. The content delivery server of Claim 15, wherein the content is
2 periodically changed based upon a presentation schedule.

1 17. The content delivery server of Claim 15, wherein the content is selected
2 from the group comprising: a content display program, a game, an entertainment
3 program, a utility program, entertainment data, advertisement data, and music data.

09648766-071800
000T70"93Z8T950


```

graph TD
    Start([START]) --> Read[Read the number of elements to be sorted]
    Read --> LoopStart(( ))
    LoopStart --> ReadElement[Read element to be sorted]
    ReadElement --> IsSmaller{Is element smaller than current element?}
    IsSmaller -- Yes --> Swap[Swap elements]
    IsSmaller -- No --> NextElement[Read next element]
    Swap --> NextElement
    NextElement --> LoopStart
    LoopStart --> End([END])
  
```



**SHIP DISK DRIVE TO OEM
INTEGRATOR, DISTRIBUTOR,
RETAILER OR CONSUMER**

INSTALL DISK DRIVE IN PERSONAL COMPUTERS	
---	--

CONFIGURE CONTENT
DELIVERY SYSTEM TO
RECEIVE USER
INFORMATION FROM
PERSONAL COMPUTERS
AND TO TRANSMIT CONTENT
DATA

PERSONAL COMPUTERS CONTACT CONTENT DELIVERY SYSTEM	
--	--

CONTENT DELIVERY
SYSTEM DELIVERS
CONTENT DATA TO
PERSONAL COMPUTERS

```

graph TD
    END([END])
  
```

FIG. 1

DISK DRIVE MANUFACTURING PROCESS

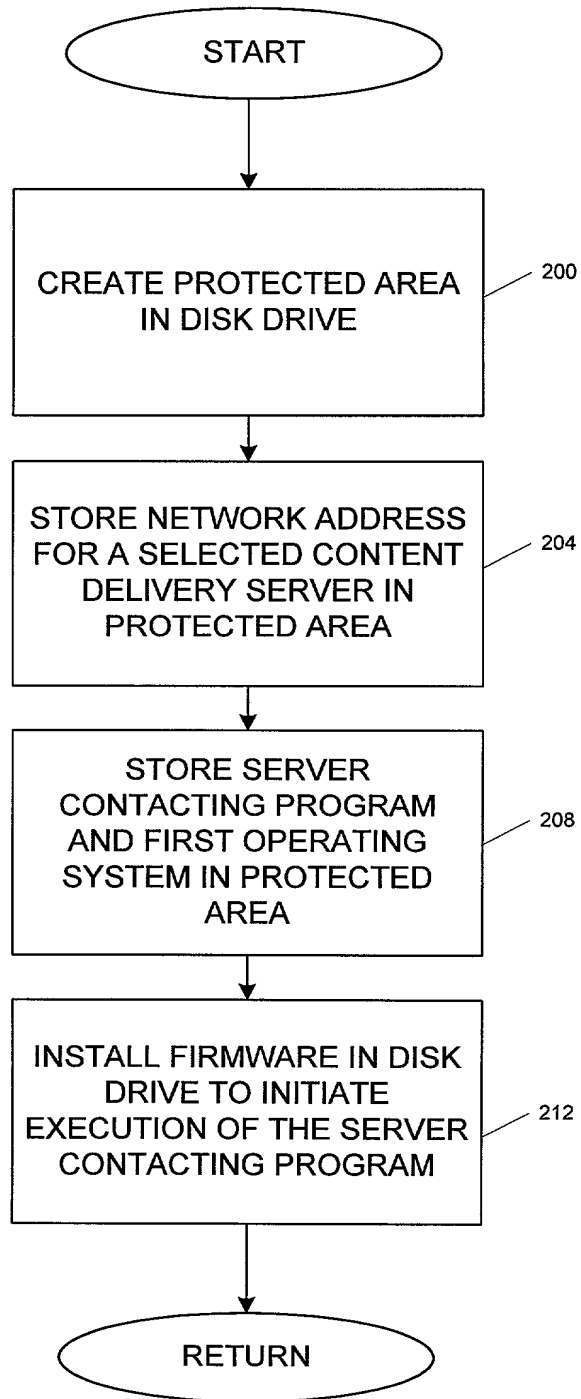


FIG. 2

INSTALLATION
PROCESS

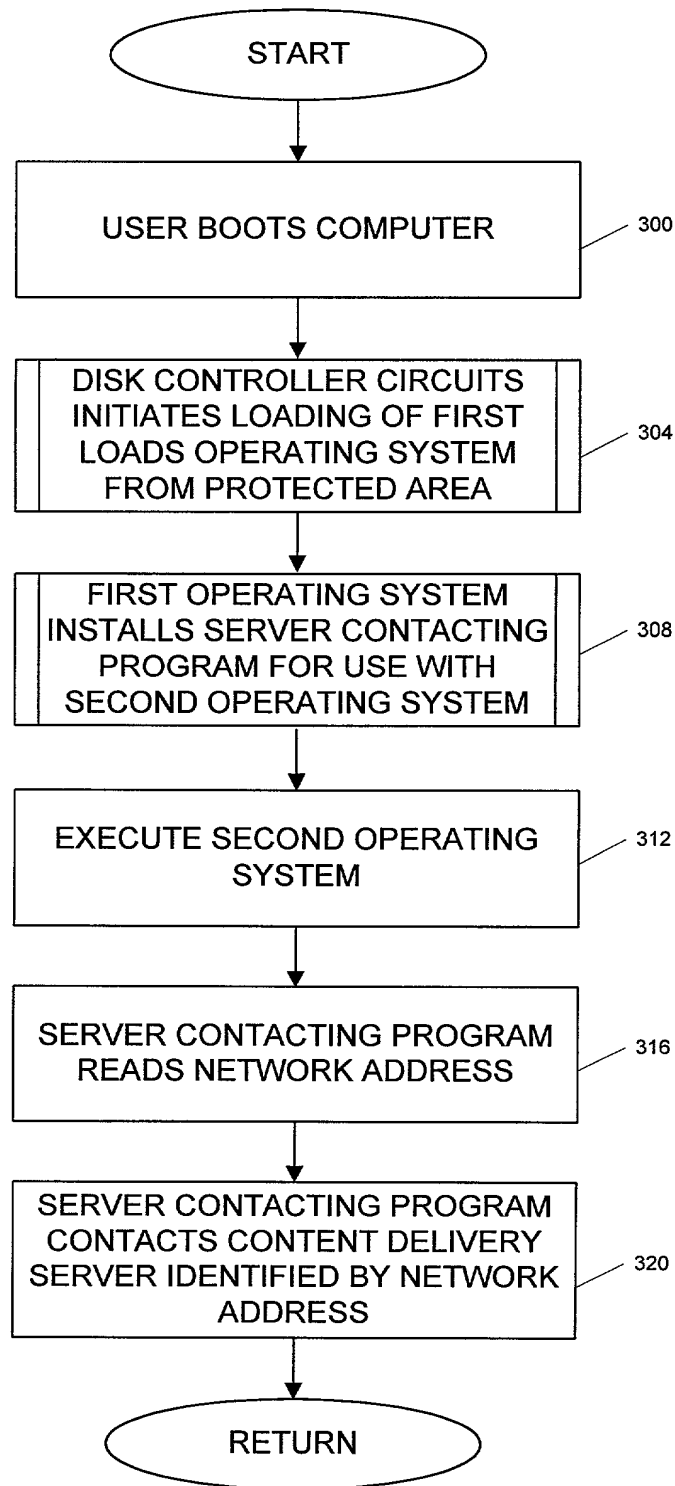


FIG. 3

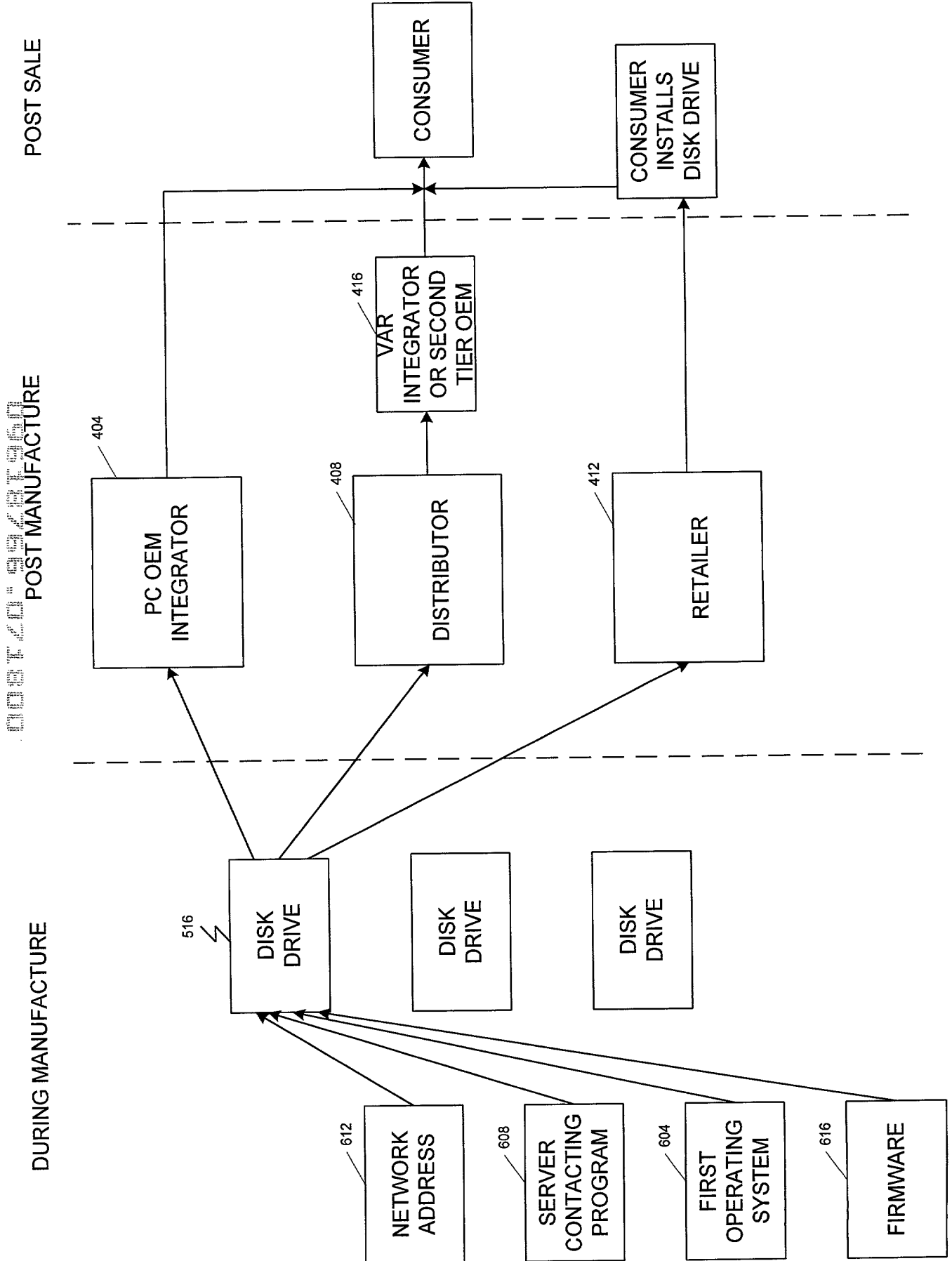


FIG. 4

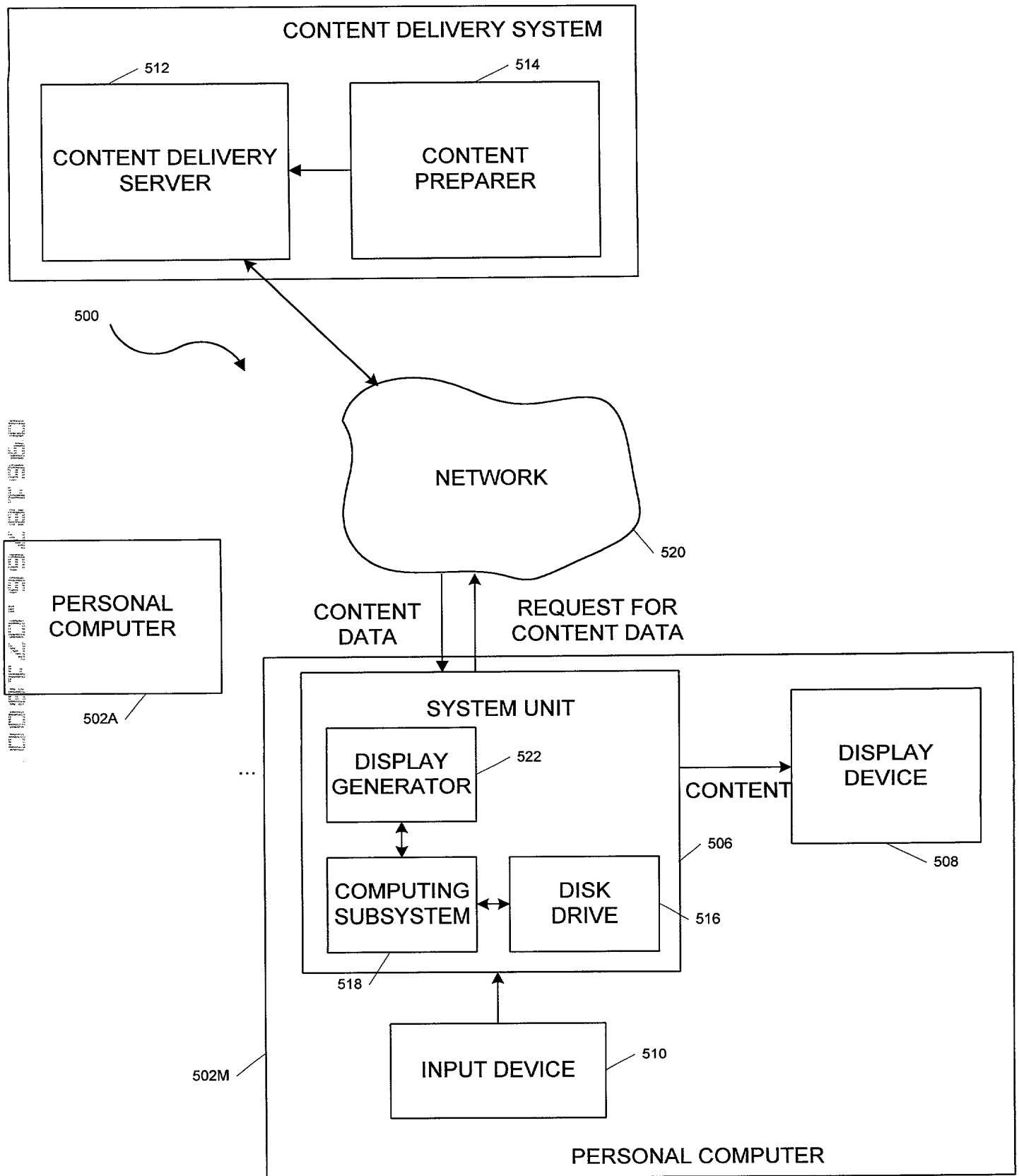


FIG. 5

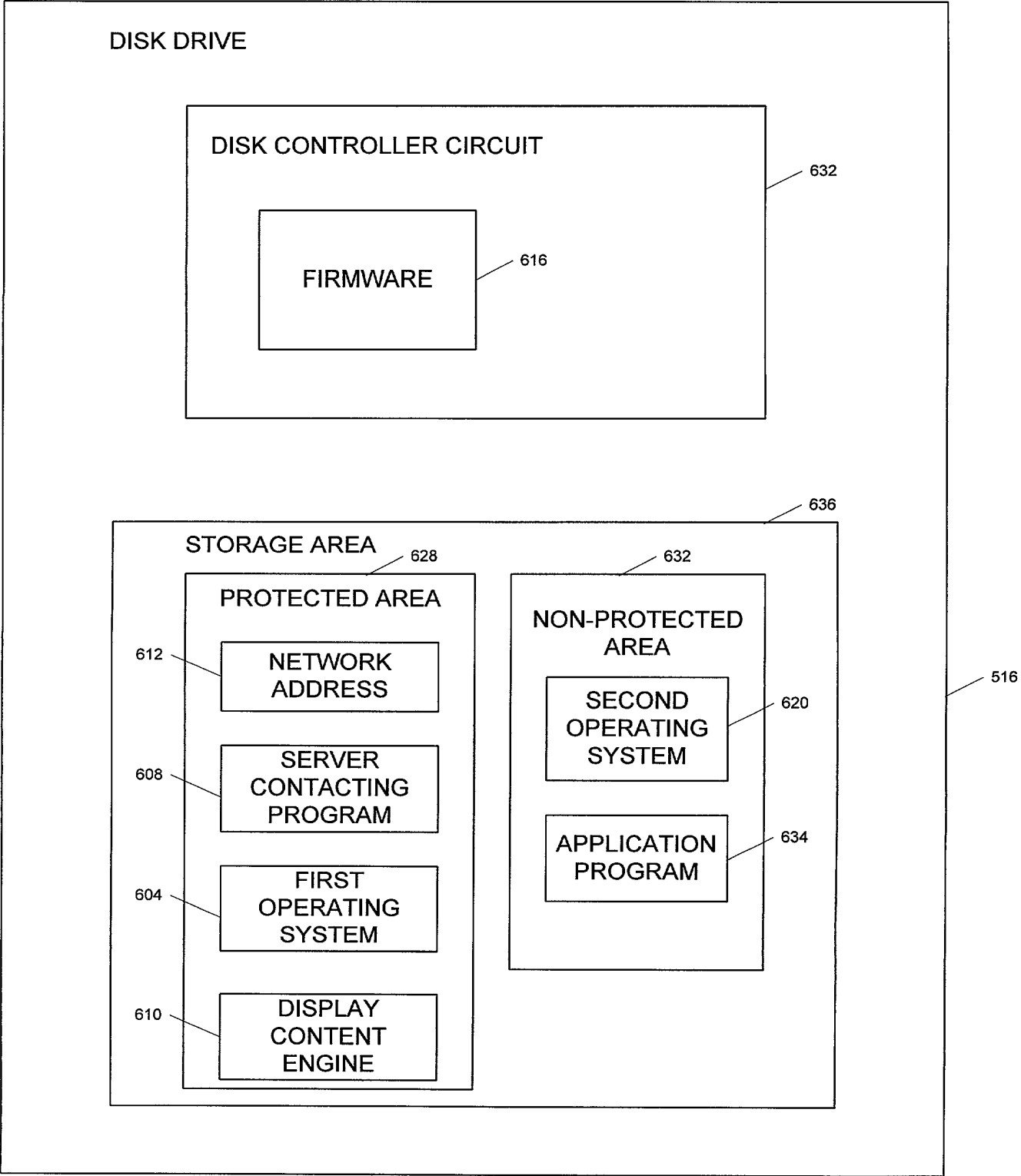


FIG. 6

Figure 1 consists of 15 bar charts, labeled (a) through (o), arranged in a grid-like fashion. Each chart displays the percentage of total protein in various fractions (A, B, C, D, E, F, G, H, I, J, K, L, M, N, O) for different protein types (A, B, C, D, E, F, G, H, I, J, K, L, M, N, O) across different conditions (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15). The charts are arranged in a grid-like fashion, with each chart having its own set of axes and labels. The data is presented as a series of bars for each fraction, with the height of the bar indicating the percentage of total protein. The charts are labeled with letters from a to o, and the protein types are labeled with letters from A to O. The conditions are labeled with numbers from 1 to 15.



CONTENT DELIVERY SYSTEM

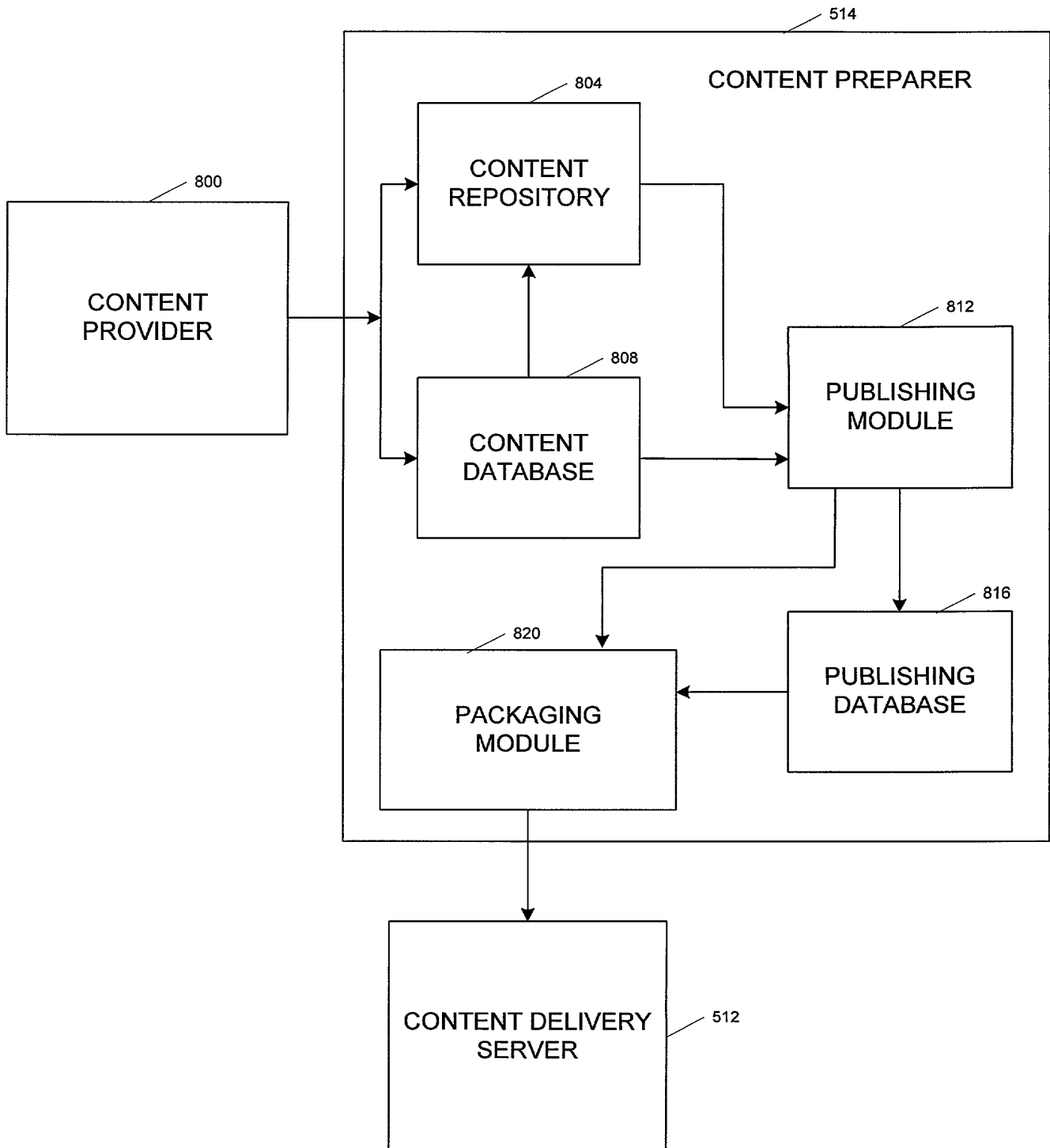


FIG. 8

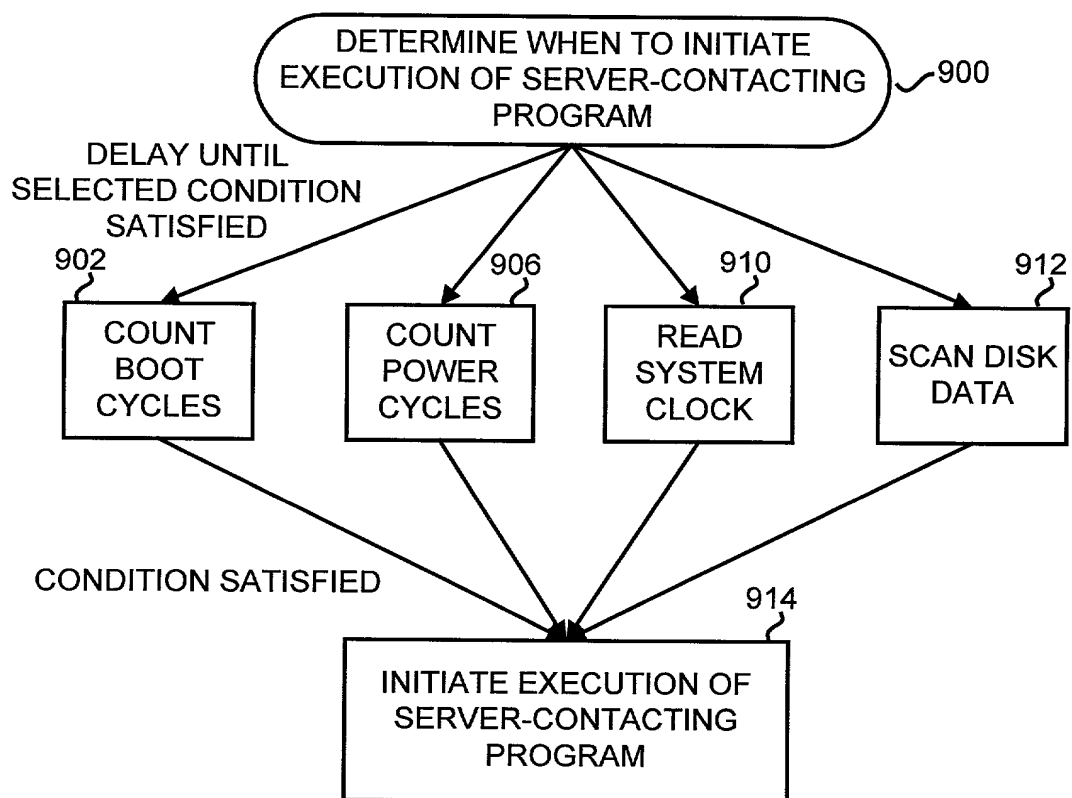


FIG. 9